



Rust For Serious Developers

Armin @mitsuhiko Ronacher

Hi, I'm Armin
...and I do Open Source

Flask
Jinja
Sentry
...



Flask

web development,
one drop at a time



SENTRY

Sentry Loves Rust

Symbolication



84.114.93.8

iPod Touch 6G
arm64iOS
Version: 10.1.1

TAGS

device | iPod Touch 6G

device.family | iPod

level | fatal

os | iOS 10.1.1

os.name | iOS

release | 1.0

Thread #0 ▾ (most recent call first)

Symbolicated

Unsymbolicated

App Only

Full

Raw

id | 0

name | n/a

was active | no

crashed | yes

EXC_BREAKPOINT

fatal error: Index out of range
Index out of range

mach exception | EXC_BREAKPOINT

SentryDemo 0x00071c7c main (AppDelegate.swift:13)

Called from: UIKit UIApplicationMain

SentryDemo 0x00070a4c @objc SentryDemo.ViewController.CrashIt (protocol<> -> () (ViewController.swift)

SentryDemo 0x00070940 SentryDemo.ViewController.CrashIt (protocol<> -> () (ViewController.swift:25)

Source Map Expansion

TypeError

Cannot read property 'id' of undefined

./app/views/projectDetails.jsx in **Constructor.onProjectChange** at line 90:59 [?](#)

```
85.     onProjectChange(projectIds) {  
86.         if (!this.state.project) return;  
87.         if (!projectIds.has(this.state.project.id)) return;  
88.  
89.         this.setState({  
90.             project: {...ProjectStore.getById(this.state.Project.id)}  
91.         });  
92.     },  
93.  
94.     identifyProject() {  
95.         let {params} = this.props;
```

Called from: ~/reflux-core/lib/PublisherMethods.js in **EventEmitter.eventHandler** [?](#)

./app/stores/projectStore.jsx in **Store.loadInitialData** at line 25:9 [?](#)



./app/views/organizationDetails.jsx in **Request.success** at line 100:21 [?](#)



./app/api.jsx in **Object.success** at line 61:20 [?](#)



Called from: ~/jquery/dist/jquery.js in **fire** [?](#)

Command Line Tool

3. mitsuhiko@herzog: ~/Development/sentry-cli (zsh) — 87×29

mitsuhiko at herzog in ~/Development/sentry-cli on git:master rust 1.15.0

\$ sentry-cli info

Sentry Server: http://localhost:8000/

Organization: default

Project: earth

Authentication Info:

Method: Auth Token

User: armin.ronacher@active-4.com (id=1)

Scopes:

- * project:read
- * project:write
- * project:delete
- * project:releases
- * team:read
- * team:write
- * team:delete
- * event:read
- * event:write
- * event:delete
- * org:read
- * org:write
- * org:delete
- * member:read
- * member:write
- * member:delete

mitsuhiko at herzog in ~/Development/sentry-cli on git:master rust 1.15.0

\$

Hello Rust

Ocaml

ADT, Pattern Matching

C++

RAII, Smart Pointers

Haskell

Typeclasses

Erlang

Message Passing, Failures

Scheme

Hygienic Macros

C#

Attributes

```
fn main() {
    println!("Hello, World!");
}
```

The Toolchain

is so good



```
curl https://sh.rustup.rs -sSf | sh
```

or on windows

```
https://win.rustup.rs
```

rustup

the rust toolchain manager

rustc

the rust compiler

cargo

the rust package manager

rustdoc

the rust documentation builder

Quickstart

```
$ cargo new --bin hello-rust
Created binary (application) `hello-rust` project

$ cd hello-rust

$ cargo run
Compiling hello-rust v0.1.0 (file:///private/tmp/hello-rust)
Finished debug [unoptimized + debuginfo] target(s)
Running `target/debug/hello-rust`
Hello, world!
```

Release Channels

stable

new release every 6 weeks

beta

upcoming stable release

nightly

includes unstable features

```
$ rustup override set nightly
info: using existing install for 'nightly-x86_64-apple-darwin'
info: override toolchain for '/private/tmp/hello-rust' set to
'nightly-x86_64-apple-darwin'
```

nightly-x86_64-apple-darwin unchanged - rustc 1.17.0-nightly
(0648517fa 2017-02-27)

The Ecosystem

crates.io

repository of reusable rust crates

docs.rs

documentation for all crates

The Features

ADTS

100

$$\int_{-1}^2 \left(\int_{x^2}^{x+2} xy dy \right) dx$$

$$2x = 4 \\ -1 > 4 - 6$$

$$(x_1, y_1) \in E_3 : [x_1, y_1] \in M_1$$

$$(a-b)(a+b) = a^2 + b^2$$

~~$$\frac{1}{2} \times 2 \times 2$$~~

$$2) \left(\frac{x}{x+2} + \frac{1}{3} = 4 \right)$$

$$2) \left(\frac{x}{x+2} + 3(x+2) \right)$$

$$3x + (x+2) = 12$$

$$P = 1 - \frac{V}{V}$$

$$3 = 12$$

$$x^2 = 12$$

$$4 \times 6 \times 2$$

-2

-4

-2

= 2

-1

0

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

```
enum JsonValue {
    Null,
    Bool(bool),
    Number(f64),
    Str(String),
    List(Box<Vec<JsonValue>>),
    Object(Box<HashMap<String, JsonValue>>),
}
```

```
fn main() {
    let value = JsonValue::Str("Hello World!".to_string());
    match value {
        JsonValue::Str(ref string) => {
            println!("JSON value as a string: {}", string);
        },
        _ => {
            println!("JSON value was something else");
        }
    }
}
```

explicit error propagation



```
use std::fs, env, io;

fn list_current_dir() -> Result<(), io::Error> {
    let here = env::current_dir()?;
    for dent_rv in fs::read_dir(here)? {
        let dent = dent_rv?;
        let md = dent.metadata()?;
        println!("{}: <{}>{}: <{}>{}",
                dent.path().display(),
                md.len(),
                if md.is_file() { "file" } else { "dir" });
    }
    Ok(())
}

fn main() {
    list_current_dir().unwrap();
}
```

strong types and inference



```
use std::fs, env, io;
use std::collections::HashMap;

fn count_extensions() -> Result<(), io::Error> {
    let mut counts = HashMap::new();
    for dent_rv in fs::read_dir(env::current_dir()...) {
        if let Some(ext) = dent_rv?.path().extension() {
            *counts.entry(ext.to_string_lossy().into_owned()).or_insert(0) += 1;
        }
    }

    let mut items = counts.into_iter().collect::<Vec<_>>();
    items.sort_by_key(|&(ext, value)| -value);
    for (ext, count) in items {
        println!("{}: >5} {}", count, ext);
    }

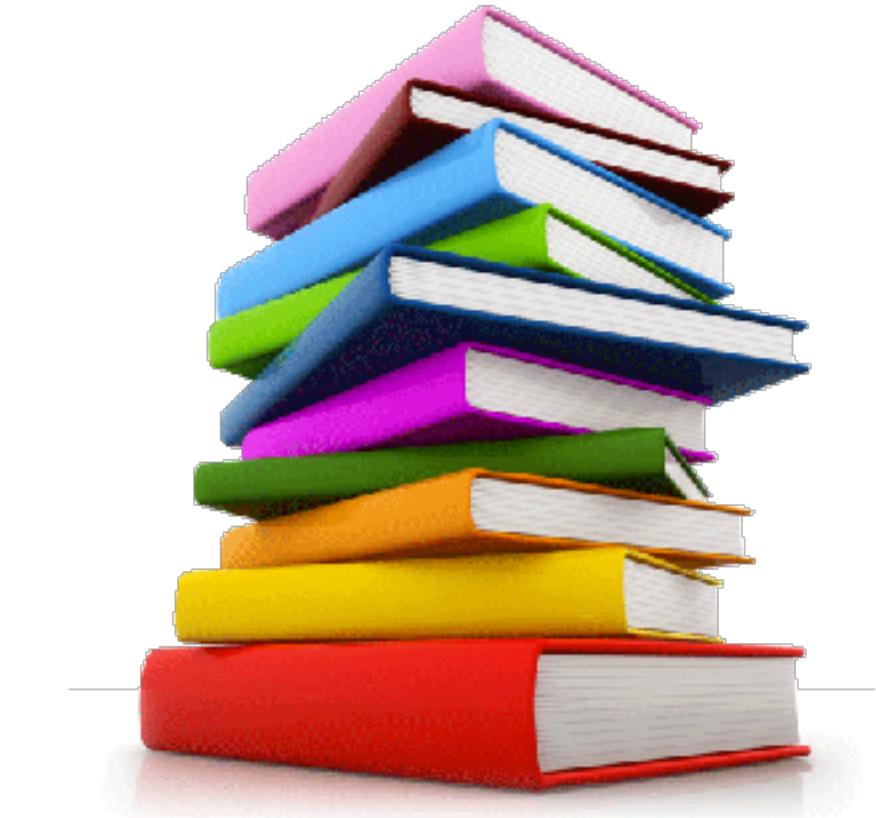
    Ok(())
}
```

Result Type

OS Str -> String

Hash Table Entry

Vector of Tuples



data vs behavior

```
struct User {
    id: u64,
    username: String,
}

impl User {
    fn new(id: u64, username: &str) -> User {
        User {
            id: id,
            username: username.to_string(),
        }
    }

    fn username(&self) -> &str {
        &self.username
    }
}
```



it's all about traits



```
use std::fmt;

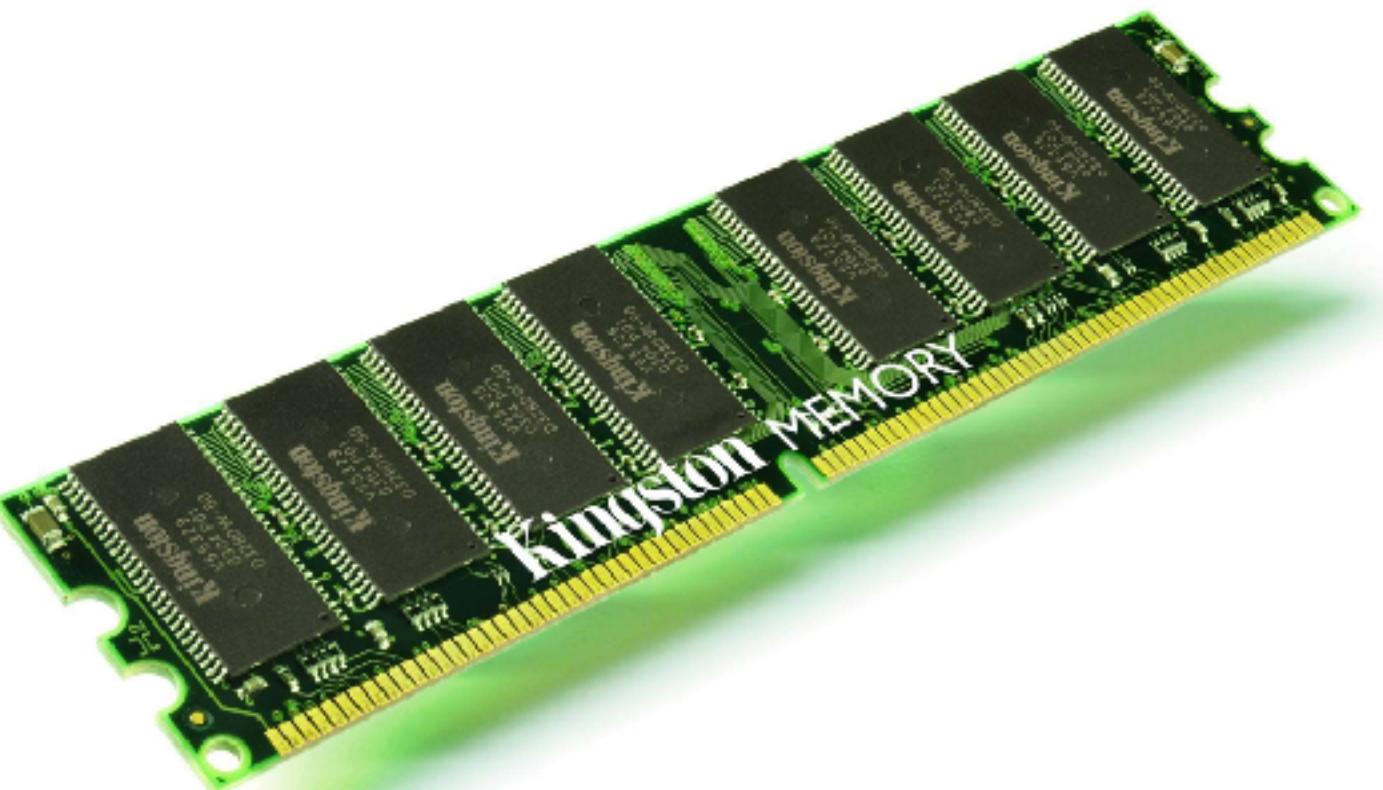
impl fmt::Debug for User {
    fn fmt(&self, f: &mut fmt::Formatter) -> fmt::Result {
        write!(f, "<User {} (id={})>", &self.username, self.id)
    }
}
```

compiler plugins



```
#[derive(Debug, Clone, Serialize, Deserialize)]
struct User {
    id: u64,
    username: String,
}
```

memory safety









so how does rust do it?

lifetimes	Reason about Memory
move semantics	Track where stuff goes
ADTs	No Null-Pointers
smart pointers	Runtime Lifetimes
bounds checking	Panic on OOB
explicit unsafe	The Escape Hatch

```
struct SdkInfo {  
    name: String,  
    version: String,  
}
```

```
struct SdkInfo<'a> {  
    name: &'a str,  
    version: &'a str,  
}
```

```
struct SdkInfo<'a> {  
    name: Cow<'a, str>,  
    version: Cow<'a, str>,  
}
```

```
fn main() {
    let items = vec![1, 2, 3];
    let sitems = items.into_iter().map(|x| x.to_string()).collect::<Vec<_>>();
    println!("Converted {} items to strings", items.len());
}
```

```
error[E0382]: use of moved value: `items`
--> fail.rs:4:46
   |
3 |     let sitems = items.into_iter().map(|x| x.to_string()).collect::<Vec<_>>();
   |           ----- value moved here
4 |     println!("Converted {} items to strings", items.len());                                ^^^^^^ value used here after move
   |
= note: move occurs because `items` has type `std::vec::Vec<i32>`,
which does not implement the `Copy` trait
```

error: aborting due to previous error

free parallelism



```
extern crate rayon;
```

Values that order & can be sent to threads

```
fn quick_sort<T: PartialOrd + Send>(v: &mut [T]) {  
    if v.len() > 1 {  
        let mid = partition(v);  
        let (low, high) = v.split_at_mut(mid);  
        rayon::join(|| quick_sort(low), || quick_sort(high));  
    }  
}
```

split mutable slice into two

Spawn two threads and join

```
fn partition<T: PartialOrd + Send>(xs: &mut [T]) -> usize {  
    let pivot = xs.len() - 1;  
    let mut i = 0;  
    for j in 0..pivot {  
        if xs[j] <= xs[pivot] { xs.swap(i, j); i += 1; }  
    }  
    xs.swap(i, pivot);  
    i  
}
```

```
struct Server {  
    ctx: Arc<Context>,  
}  
  
impl Server {  
    pub fn run(&self) -> Result<()> {  
        let ctx = self.ctx.clone();  
        Server::http("127.0.0.1:3000")?  
            .handle(move |req: Request, resp: Response|  
        {  
            let local_ctx = ctx.clone();  
            ...  
        })  
    }  
}
```

increments refcount and returns new Arc

Should you use it?

maybe!

Python Modules

Redis/Nginx Modules

Command Line Tools

Native Libraries

One-Trick-Pony API Servers

Our Experience

code is very readable

compile times suck

crate ecosystem is great

highly maintainable

Q&A